LA-UR- 01-4336

Approved for public release;
distribution is unlimited.

Title:

An Approach to Extreme-Scale Simulation of Novel Architectures

Author(s):

Francis J. Alexander
Kathryn P. Berkbigler
Graham Booker
Brian W.Bush
Kei Davis
Adolfy Hoisie

Submitted to:

Fourth Biennial Tri-Laboratory Engineering
Conference on Modeling and Simulation,
DOE Energy Training Center,
Albuquerque, New Mexico,
23-24 October 2001

An Approach to Extreme-Scale Simulation of Novel Architectures

F. Alexander, K. Berkbigler, G. Booker, B. Bush, K. Davis, A. Hoisie

Los Alamos National Laboratory

Presentation to Fourth Biennial Tri-Laboratory Engineering Conference on Modeling and Simulation

23 October 2001

NNSA

LA-UR-01-4336

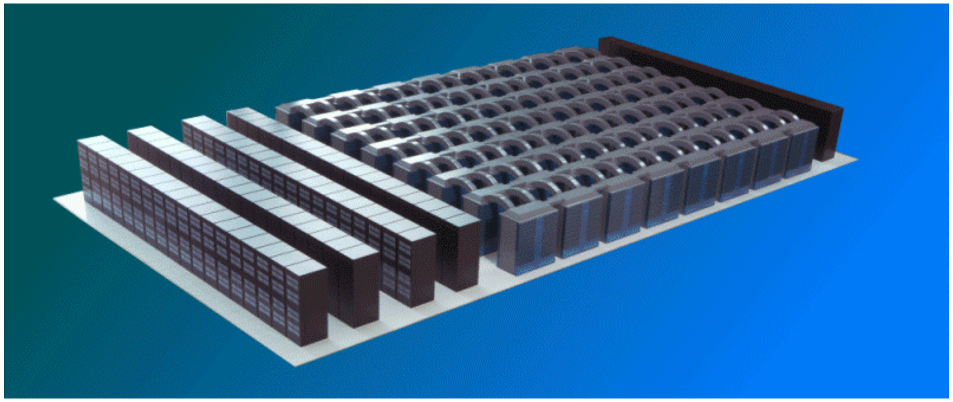
Los Alamos

Abstract

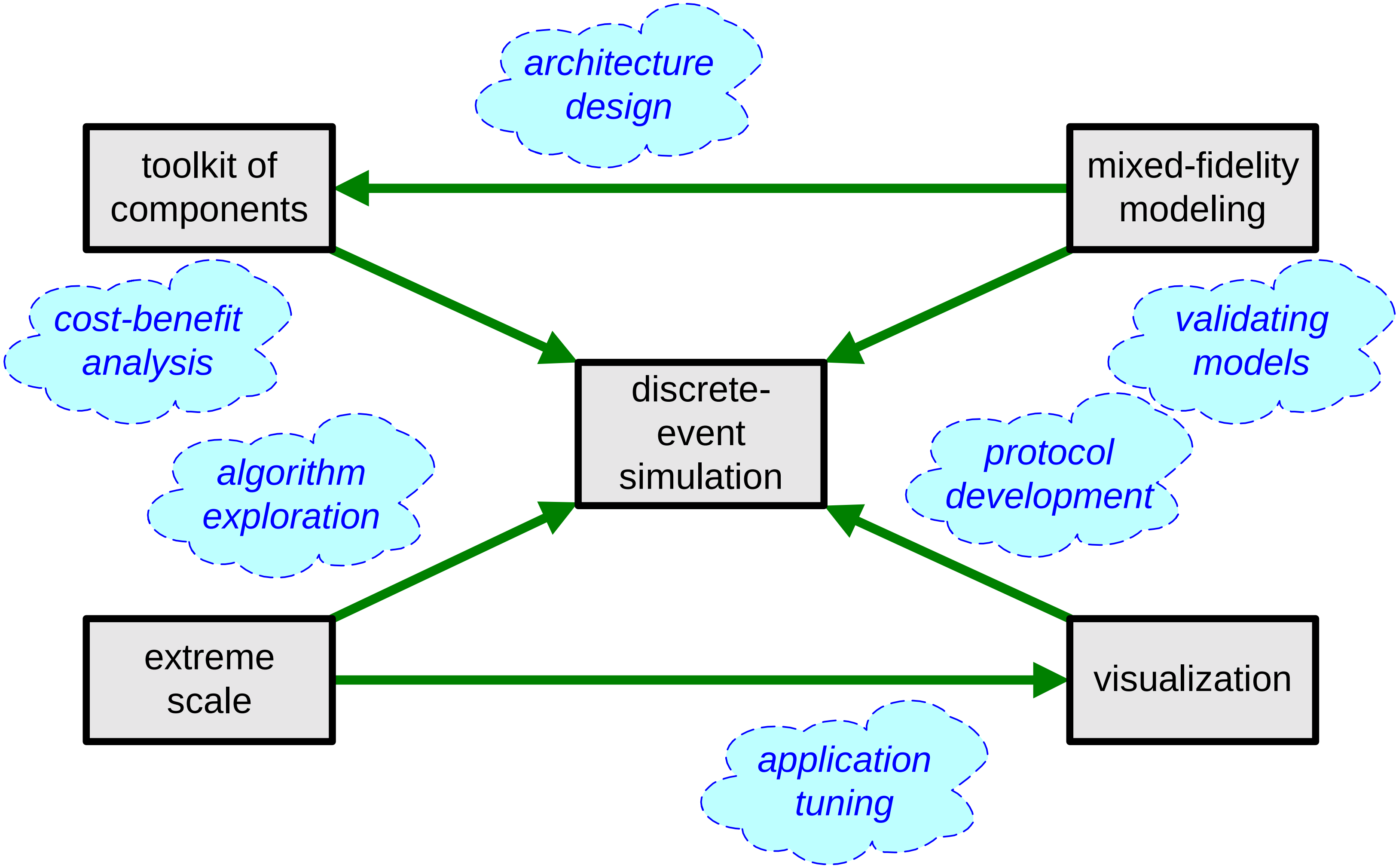
The magnitude of the scientific computations targeted by the ASCI project requires as-yet unavailable computational power. To facilitate these computations ASCI plans to deploy massive computing platforms, possibly consisting of tens of thousands of processors, capable of achieving 10-100 tera-ops. For various reasons the current approach to building a yet-larger supercomputer—connecting commercially available SMPs with a network—may be reaching practical limits. The path to better hardware design and lower development costs involves performance evaluation, analysis, and modeling of parallel applications and architectures, and in particular predictive capability. We outline an approach for simulating computing architectures applicable to extreme-scale systems (thousands of processors) and to advanced, novel architectural configurations. The proposed simulation environment could be used for: (i) exploration of hardware/architecture design space; (ii) exploration of algorithm/implementation space both at the application level (e.g. data distribution and communication) and the system level (e.g. scheduling, routing, and load balancing); (iii) determining how application performance will scale with the number of processors or other components; (iv) analysis of the tradeoffs between performance and cost; (v) testing and validating analytical models of computation and communication. Our component-based design allows for the seamless assembly of architectures from representations of workload, processor, network interface, switches, etc., with disparate resolutions, into an integrated simulation model. This accommodates different case studies that may require different levels of fidelity in various parts of a system. Our initial prototype, comprising low-fidelity models of workload and network, aims to model at least 4096 computational nodes in a fat-tree network. It supports studies of simulation performance and scaling rather than the properties of the simulated system themselves. Future work will allow more realistic simulation and visualization of ASCI-like workloads on very large machines.

Context

- *The magnitude of the scientific computations targeted by the ASCI project requires as-yet unavailable computational power.*
- *Current approaches to building larger supercomputers—connecting commercially-available SMPs with a network—may be reaching practical limits.*
- *In response, the DOE Advanced Architecture Initiative seeks to research alternative high-performance computing architectures.*
- *The path to better hardware design and lower development costs involves performance evaluation, analysis, and modeling of parallel applications and architectures, and in particular predictive capability.*



Goals



http://www.c3.lanl.gov/~parsim

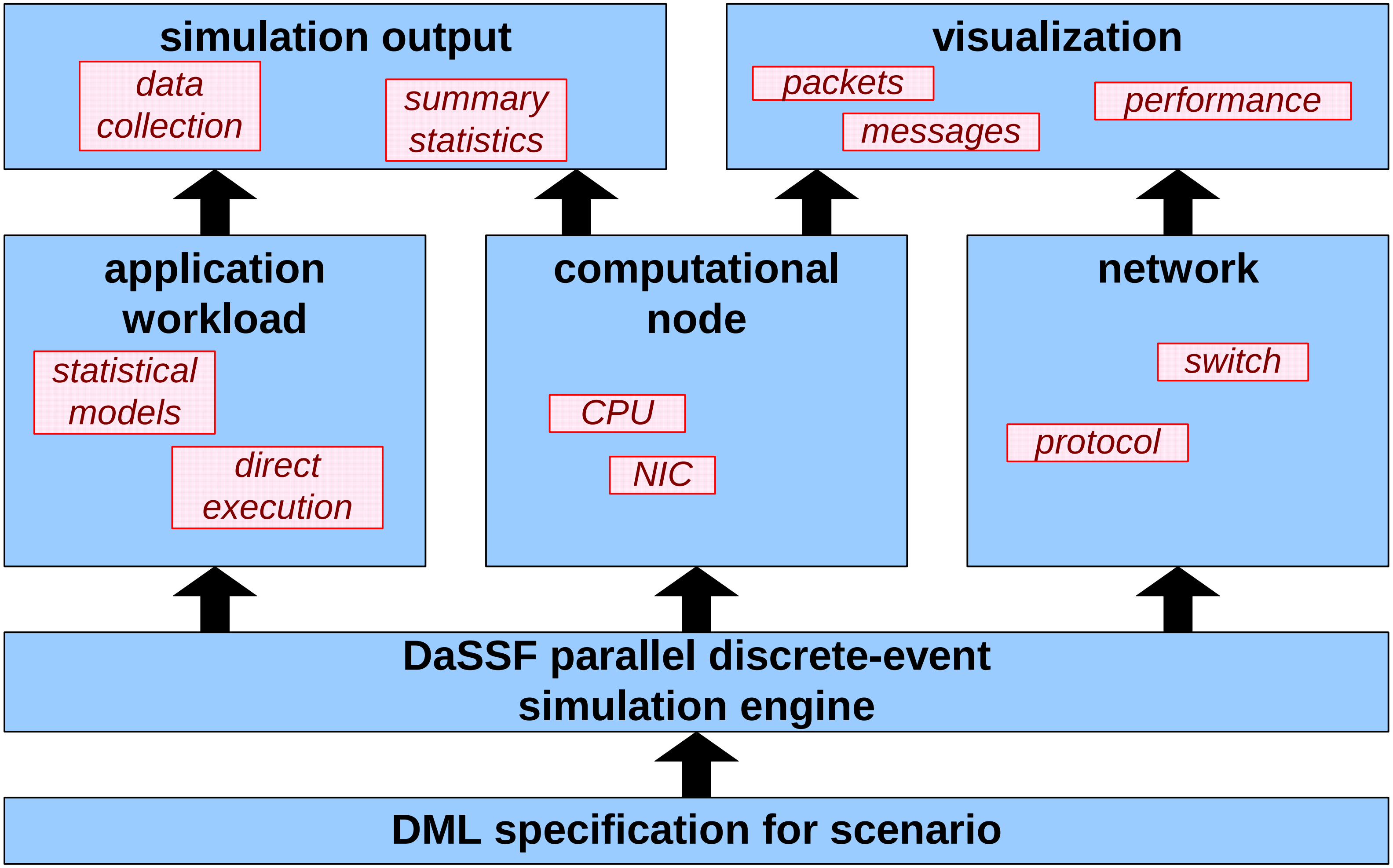
Approach

- *iterative development*
- *portable implementation*
- *efficient parallel discrete-event simulation*
  - *scalable to thousands of computational nodes*
- *component-based design*
  - *workloads*
  - *processors*
  - *network interfaces*
  - *switches*
- *multiple-fidelity representations*
  - *mix & match components of different fidelities*
  - *construct model with appropriate level of detail for a study*
- *seamless assembly of architectures*
- *visualization*

Workload Representation

- *Simple random processes can load the hardware with message traffic having specified statistical properties.*
  - *matches distribution of messages*
  - *can include temporal and spatial correlations between messages*
  - *ignores some of the data dependency*
- *Direct-execution techniques allow one to run programs nearly exactly on real processors coupled to a simulated network.*
  - *is faithful to actual timing on processors*
  - *may be too computationally intensive or slow*
- *From the time series of fine-grained simulations we will use learning algorithms to construct reduced models of the full system dynamics.*
  - *uses regression techniques like neural networks or dimension reduction methods such as the Karhunen-Loeve expansion*

Simulation Architecture



http://www.c3.lanl.gov/~parsim

DaSSF: Dartmouth Scalable Simulation Framework

- *conservative discrete-event simulation*
  - *handles synchronization and scheduling*
- *lean C++ API*
  - `Entity`
  - `Process`
  - `inChannel`
  - `outChannel`
  - `Event`
- *parallel*
  - *shared memory*
  - *distributed memory (MPI)*
- *scalable*
  - *lightweight custom threads*
- *multiple platforms*

Model Specification

- *DML: Domain Modeling Language*
  - *recursively-defined list of attributes*
  - *supports composition and “inheritance”*
  - *text format*
- *easy to construct library of reusable component specifications*
- *partitioner decomposes model for later parallel computation*

DML ::= attribute-list

attribute-list ::= empty | attribute-list attribute

attribute ::= key value | key [attribute-list]

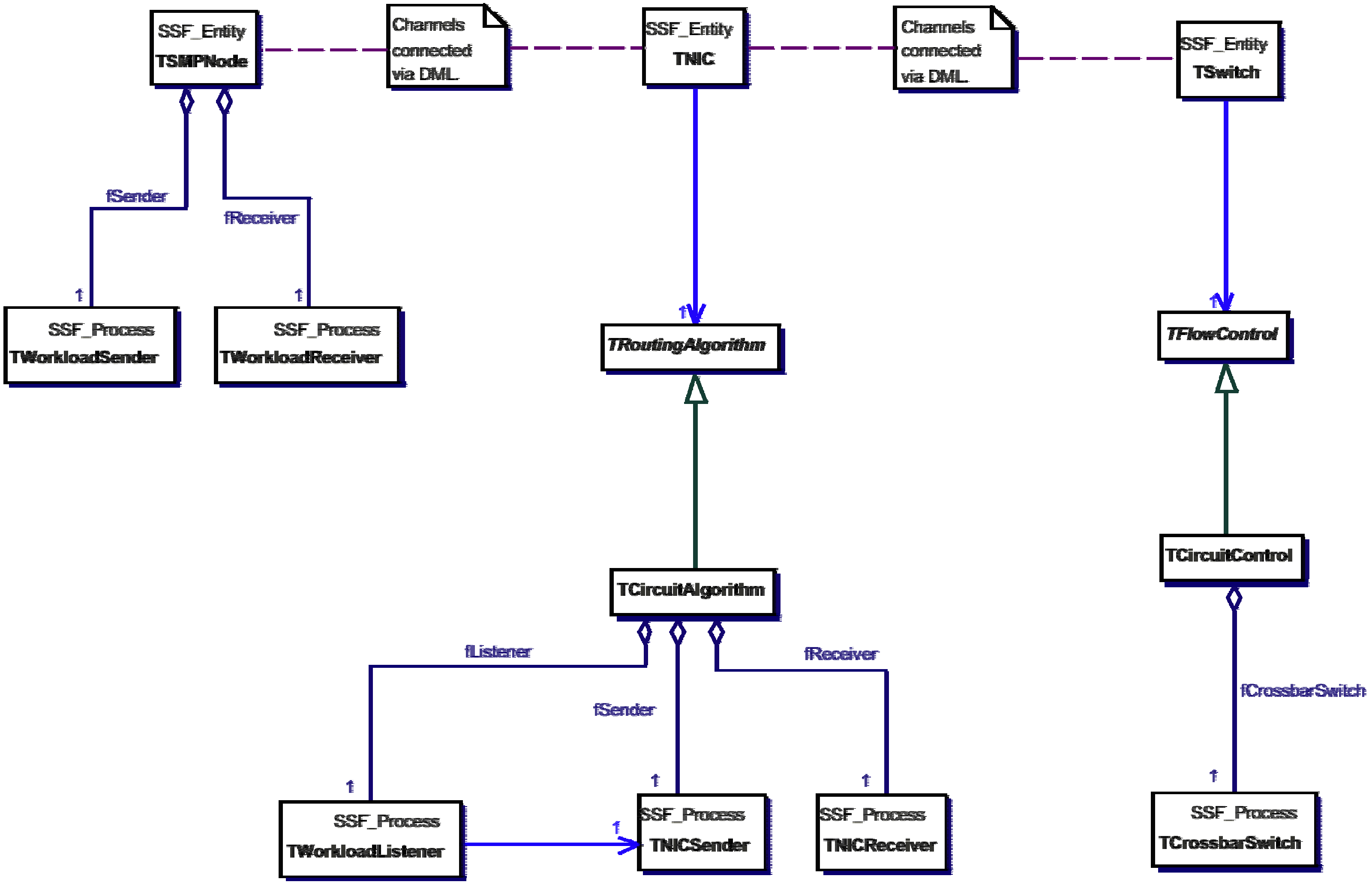
key ::= [a-zA-Z_][a-zA-Z0-9]*

value ::= INTEGER | FLOAT | STRING

```
MODEL [
  CLUSTER [
    ID 0
    ENTITY [ _extends .parsim.node
      CONFIGURE [
          TARGET "1”
      ]
    ]
    ENTITY [ _extends .parsim.nic
      PARAMS [
        STRING "CircuitAlgorithm”
      ]
      CONFIGURE [
        ROUTE [TARGET "2” PATH [PORT 5 PORT 3]]
      ]

    ]
    MAP [FROM 0(BUSOUT) TO 1(BUSIN) DELAY 1]
    ALIGN [FROM 0 TO 1]
  ]
. . .
```

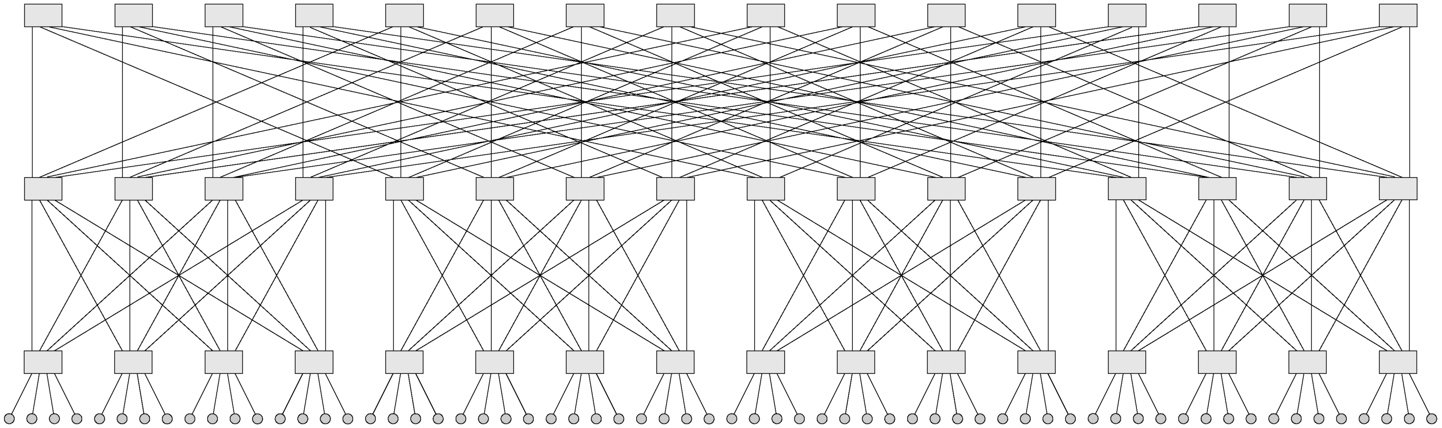
Simplified UML Class Diagram



http://www.c3.lanl.gov/~parsim

Prototype Models

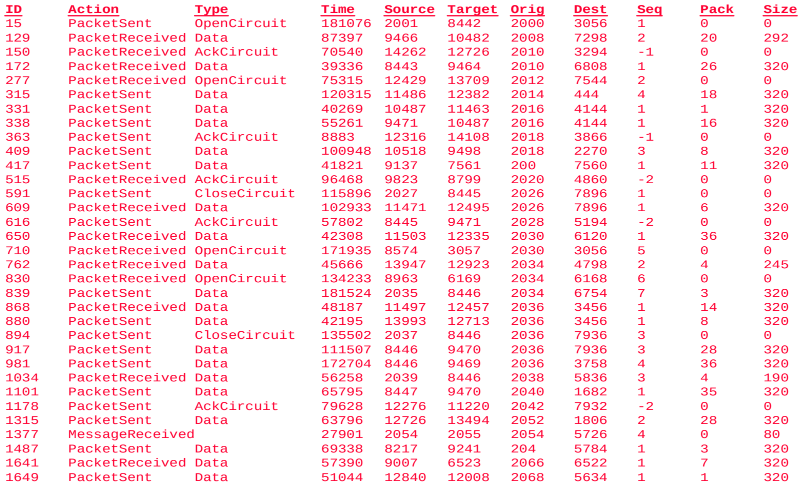
- *compute nodes with simple statistical model of workload*
  - *exponentially-distributed message sizes and spacings*
  - *configurable source-target patterns for messages*
- *circuit-switched "fat"-tree network*
  - *switches with four "up" ports and four "down" ports*
  - *packet-level resolution*
- *configurable time delays between components*
- *sample models: 64 (below) or 4096 compute nodes (6 layers)*



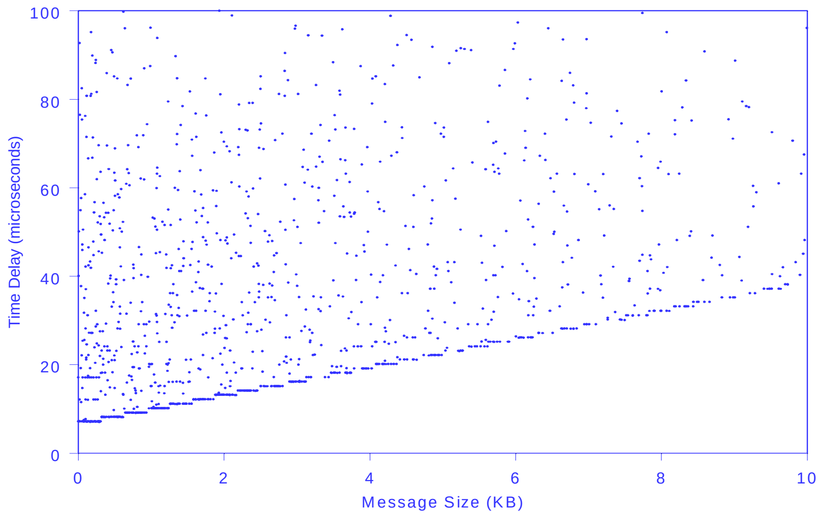
http://www.c3.lanl.gov/~parsim

Simulation Output

- complete detail
  - history of messages
  - propagation of packets
- summaries
  - sliced into time windows
  - queue sizes
  - throughputs
  - port usages
  - timeouts
  - path lengths
  - communication patterns
- configurable
  - time ranges
  - choice of output
- text format



```
ID      Action         Type           Time   Source  Target Orig    Dest   Seq     Pack   Size
15      PacketSent     OpenCircuit    181076 2001    8442   2000    3056   1       0      0
129     PacketReceived Data           87397  9466    10482  2008    7298   2       20     292
150     PacketReceived AckCircuit     70540  14262   12726  2010    3294   -1      0      0
172     PacketReceived Data           39336  8443    9464   2010    6808   1       26     320
277     PacketReceived OpenCircuit    75315  12429   13709  2012    7544   2       0      0
315     PacketSent     Data           120315 11486   12382  2014    444    4       18     320
331     PacketSent     Data           40269  10487   11463  2016    4144   1       1      320
338     PacketSent     Data           55261  9471    10487  2016    4144   1       16     320
363     PacketSent     AckCircuit     8883   12316   14108  2018    3866   -1      0      0
409     PacketSent     Data           100948 10518   9498   2018    2270   3       8      320
417     PacketSent     Data           41821  9137    7561   200     7560   1       11     320
515     PacketReceived AckCircuit     96468  9823    8799   2020    4860   -2      0      0
591     PacketSent     CloseCircuit   115896 2027    8445   2026    7896   1       0      0
609     PacketReceived Data           102933 11471   12495  2026    7896   1       6      320
616     PacketSent     AckCircuit     57802  8445    9471   2028    5194   -2      0      0
650     PacketReceived Data           42308  11503   12335  2030    6120   1       36     320
710     PacketReceived OpenCircuit    171935 8574    3057   2030    3056   5       0      0
762     PacketReceived Data           45666  13947   12923  2034    4798   2       4      245
830     PacketReceived OpenCircuit    134233 8963    6169   2034    6168   6       0      0
839     PacketSent     Data           181524 2035    8446   2034    6754   7       3      320
868     PacketReceived Data           48187  11497   12457  2036    3456   1       14     320
880     PacketSent     Data           42195  13993   12713  2036    3456   1       8      320
894     PacketSent     CloseCircuit   135502 2037    8446   2036    7936   3       0      0
917     PacketSent     Data           111507 8446    9470   2036    7936   3       28     320
981     PacketSent     Data           172704 8446    9469   2036    3758   4       36     320
1034    PacketReceived Data           56258  2039    8446   2038    5836   3       4      190
1101    PacketSent     Data           65795  8447    9470   2040    1682   1       35     320
1178    PacketSent     AckCircuit     79628  12276   11220  2042    7932   -2      0      0
1315    PacketSent     Data           63796  12726   13494  2052    1806   2       28     320
1377    MessageReceived               27901  2054    2055   2054    5726   4       0      80
1487    PacketSent     Data           69338  8217    9241   204     5784   1       3      320
1641    PacketReceived Data           57390  9007    6523   2066    6522   1       7      320
1649    PacketSent     Data           51044  12840   12008  2068    5634   1       1      320
```

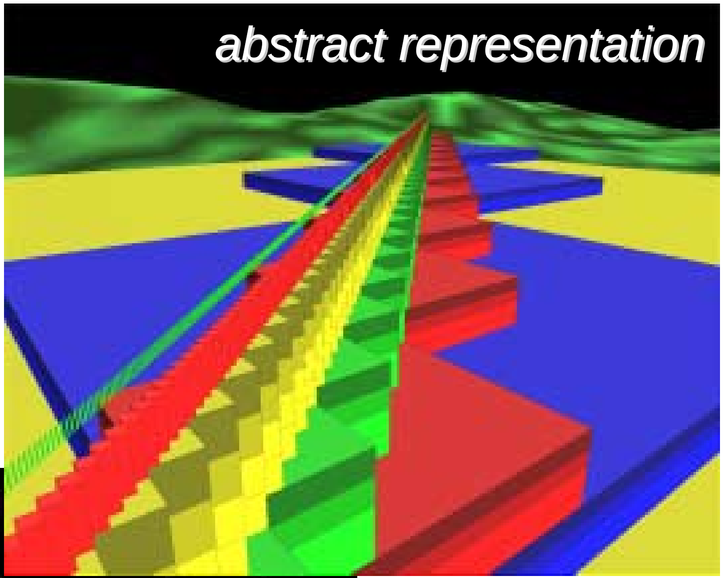


http://www.c3.lanl.gov/~parsim

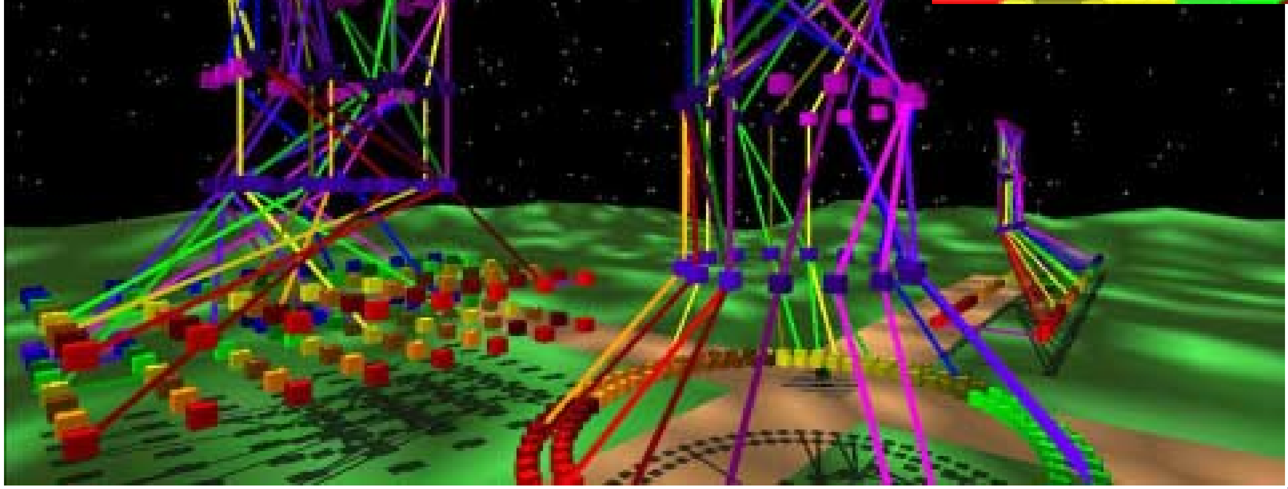
*Visualization**

- *simulation debugging*
- *network behavior & performance*
- *application communication patterns and network usage*

**in collaboration with Tom Caudell* et al. *of the Albuquerque High Performance Computing Center at the University of New Mexico*



abstract representation



direct representation

http://www.c3.lanl.gov/~parsim

Performance & Portability

- *platform requirements*
  - *posix*
  - *DaSSF (uses MPI)*
- *depends on workload representation*
  - *direct execution of workload may limit portability*
  - *communication-bound for statistical workload models*
- *have already simulated 4096 nodes, 6144 switches: 22 simulated nanosecs per cpu-sec*
- *not optimized yet!*

Test Simulations of 64 Nodes

Platform	Comp. Nodes	Events (1/sec)	Execution Time (sec)
Linux, 733 MHz Pentium III	1	2020.2	178.1
	2	689.9	521.6
Linux, 500 MHz Pentium III	4	494.3	728.0
	8	379.9	947.3
Solaris, Sun SPARC Ultra 5	1	1105.1	325.6
Irix, Origin 2000	1	1298.3	277.2
	4	1351.1	266.4
	16	630.2	571.1

http://www.c3.lanl.gov/~parsim

Challenges & Risks

- *scaling*
  - *simulating extreme scale systems is resource-intensive even with efficient simulators*
- *fidelity*
  - *constructing high-fidelity models of processors or networks is a labor-intensive process*
  - *modeling operating system behavior might be required in some studies*
- *portability*
  - *accurately measuring directly-executed applications requires non-portable timers*
  - *modeling of low-level networking APIs may reduce portability*
- *validation*
  - *detailed measurements of applications on large machines are needed to validate a simulation*

Applications

- *exploration of hardware/architecture design space*
  - *novel architectures*
- *exploration of algorithm/implementation space*
  - *application level*
    - *data distribution*
    - *communication*
  - *system level*
    - *scheduling*
    - *routing*
    - *load balancing*
- *determining how application performance will scale with number of processors or other components*
- *analysis of tradeoffs between performance and cost*
- *testing and validating analytical models of computation and communication*

Status

- completed since April
  - low-fidelity prototype
  - simulation engine (DaSSF) evaluation
  - initial simulation performance studies on 4096 compute nodes
- ongoing work
  - simulation performance and scaling
  - realistic network protocols and timings
  - direct-execution-based workload
    - MPI applications (starting with "sweep3d")
  - proof-of-principle for reduced models of workload
  - visualization
- future directions
  - validation study
  - I/O and storage devices
  - wide-area networking